

What is claimed is:

1. A method of transferring application-generated structured data from a client to a server, such that information structures specified by the data can be automatically expanded into corresponding information structures displayable by a website in communication with the server, the method including the steps of:

providing an interface at the client for accepting application-generated structured data,  
converting into a source-independent format the accepted application-generated structured data, thereby to generate a source-independent data package,  
transmitting to the server the source-independent data package,  
receiving at the server the source-independent data package, and  
creating or modifying on the server, in accordance with the received source-independent data package, information structures corresponding to the information structures specified by the application-generated structured data.

2. The method of claim 1 wherein the structured data contains embedded information providing description about organization, hierarchical decomposition or source of the data.

3. The method of claim 2 wherein the structured data is CAD data.

4. The method of claim 3 wherein the structured data contains embedded information providing description about organization of a three-dimensional design into assemblies, sub-assemblies and parts.

5. The method of claim 4 wherein the CAD data is created by a CAD software tool.

6. The method of claim 5 wherein the CAD software tool is Solidworks or ProEngineer.
7. The method of claim 2 wherein the structured data is representative of a project schedule.
8. The method of claim 7 wherein the structured data is produced by Microsoft Project
9. The method of claim 8 wherein the structured data includes information about tasks and sub-tasks of the schedule.
10. The method of claim 2 wherein the structured data is produced by Microsoft PowerPoint.
11. The method of claim 2 wherein the structured data is produced by Microsoft Excel.
12. The method of claim 1 wherein the client runs a process to determine which of a plurality of document handlers is to be used to process the structured data.
13. The method of claim 1 wherein the converting step includes the step of interpreting the structured data and creating an ApXML file representative of the application-generated structured data.
14. The method of claim 13 wherein the interpreting step is executed by the client, which runs a document handler that interprets the structured data in the designated file directly or by opening the creating application and interrogating the creating application through an API to obtain the structured data.

15. The method of claim 14 wherein, following interpretation of the structured data, the client packages all relevant data, including ApXML and graphics files or other information items, and then causes the package to be transferred to the server.
16. The method of claim 15 wherein the server, following receipt of the package, interprets the ApXML and in response to the interpretation of the ApXML and supporting files, creates or modifies information structures on the website in accordance with information contained in the ApXML and supporting files.
17. The method of claim 16 wherein the client contains an Inbox Control comprising an ActiveX control that functions as a “drop” target in a GUI image presented by the website.
18. The method of claim 1 wherein the client contains an Inbox Control comprising an ActiveX control that functions as a “drop” target in a Graphical User Interface image presented by the website, and the Inbox Control functions to accept the “dropped” data.
19. The method of claim 18 wherein the client contains a Data Selector that controls bidding and selection of an appropriate data handler.
20. The method of claim 19 wherein the Data Selector employs restrictions in the bidding process to restrict selected types of data.
21. The method of claim 20 wherein the client contains at least one Data Client that functions as a data handler.
22. The method of claim 18 wherein the client contains a plurality of Data Clients that (1) collectively constitute an extendible set of data handlers, (2) bid for the handling of data types they recognize, (3) can be selected by the Data Selector, and (4) once selected by the Data Selector, process the data.

23. The method of claim 22 wherein new Data Clients can be added and registered with the system and bidding process, whereby the set of Data Clients is extended.
24. The method of claim 23 wherein the client contains a Bundler Control that bundles into discrete files any data designated for transfer to the server.
25. The method of claim 24 wherein the server contains an Active Manager that unbundles data received from the client.
26. The method of claim 25 wherein the server contains a Data Processor that controls the manner in which “dropped” data are applied to the website in accordance with information contained in the data.
27. The method of claim 26 wherein the server contains a plurality of Data Processors and the Active Manager selects an appropriate Data Processor from among the plurality of Data Processors to process the received data.
28. The method of claim 27 wherein the server contains an ApXML processor that processes the received XML data and creates website structure according to instructions from the selected Data Processor.
29. The method of claim 28 wherein the server contains a Bundler Control that is employed to un-bundle the bundled files received from the client.
30. The method of claim 29 further comprising the following steps:
  - navigating, using a web browser on the client, to a web page providing an interface to the Inbox Control,
  - dragging and dropping, on a GUI supported by the web browser, at least one document containing structured data onto the display area associated with the Inbox Control,
  - passing the structured data to the Data Selector,

determining, using the Data Selector, the identities of registered Data Clients,  
passing the structured data to the registered Data Clients,  
receiving bids from the Data Clients, responsive to each Data Client's  
examination of the structured data,  
evaluating, based upon predetermined criteria, the bids received from the Data  
Clients,  
selecting, responsive to the evaluating step, a Data Client to process the structured  
data,  
processing, using the selected Data Client, the structured data, and  
creating or modifying, on the server, information structures specified by the  
structured data, thereby to create or modify structure on the website.

31. The method of claim 30 wherein the processing step includes the step of  
interrogating the file and extracting therefrom information descriptive of structure,  
graphic view and any associated predefined information items extracted from the original  
source data file, project element thumbnail images, preview images or hotspots.
32. The method of claim 31 wherein the interrogating step includes either interpreting  
the file data directly or calling the file's creating application and using an API associated  
with the application.
33. The method of claim 32 wherein the processing step includes the further step of  
activating a user interface to obtain information from the user.
34. The method of claim 33 wherein the information can include options affecting  
structure handling or geographical representations.
35. The method of claim 34 wherein the processing step further includes using  
information about the document structure to create an ApXML file describing hierarchy  
and related structured data.

36. The method of claim 35 wherein the processing step further includes placing in a designated directory structure all corresponding files.

37. The method of claim 36 comprising the further step of returning, using the Data Client, a list of files and associated information back to the Data Selector.

38. The method of claim 37 wherein associated information includes Process Type, Document Type and Document Subtype.

39. The method of claim 38 comprising the further step of passing, using the Data Selector, all associated information back to the Inbox Control.

40. The method of claim 39 comprising the further step of utilizing the ApXML Processor to create or modify information structures specified by the structured data, in response to the APXML and other files received from the client.

41. The method of claim 40 comprising the further step of transmitting to a website in communication with the server, events that cause the website to be updated and regenerated in correspondence with the application-generated structured data generated at the client.

42. The method of claim 41 wherein data processing restrictions can be assigned to the structured data at the Inbox Control.

43. The method of claim 42 wherein bids are based upon predetermined criteria.

44. The method of claim 43 wherein the criteria can include file extension, file content or restrictions.

45. The method of claim 44 wherein bids are generated in accordance with a Data Client's evaluation of its ability to handle a given structured data file.

46. The method of claim 45 wherein the greater the Data Client's ability to handle the file, the higher the Data Client's bid.

47. The method of claim 46 wherein the structured data file is passed to the highest bidder.

48. The method of claim 47 wherein the Inbox Control is instantiated and initialized using HTML and Java Script in the web page containing the interface to the Inbox Control.

49. The method of claim 48 wherein ApXML functions as a mechanism to import any structured data in a manner independent of the source of that data.

50. The method of claim 49 wherein ApXML is a specific definition of XML tags used to represent an ActiveProject internal meta-design structure.

51. The method of claim 50 wherein the ActiveProject meta-design describes aspects of a design, including categories of information, the hierarchical decomposition of each aspect into project elements and sub-project elements, and any properties and associated information items associated with each project element.

52. The method of claim 51 wherein the properties and information items associated with each project element can include files, URLs, and database queries.

53. The method of claim 52 wherein the meta-design also describes a graphic appearance for each project element.

54. The method of claim 53 wherein the graphic appearance can include a thumbnail image or a hotspot over a background picture.

55. The method of claim 54 wherein each element in the meta-design has a corresponding tag in ApXML.
56. The method of claim 55 wherein a tag in ApXML is used to designate any information items or files associated with an aspect of a project.
57. The method of claim 56 wherein information items can include name, description, document type and file path.
58. The method of claim 57 wherein ApXML supports definitions of top level aspects of a project or design.
59. The method of claim 58 wherein top level aspects of the project or design are represented as tabs in the associated web sites.
60. The method of claim 59 wherein ApXML supports definition of project elements and sub-elements using XML tags.
61. The method of claim 60 wherein ApXML supports definition of project element properties, including associated thumbnail images, and association of predefined information item files with the defined project element.
62. The method of claim 61 wherein ApXML supports definition of a reference identifier so that when the original data files are republished, the same object can be recognized.
63. A method for replicating, on a website in communication with a server, representations of application-specific structured data originating on a client, the method comprising the steps of:



obtaining application-specific data at the client, where the application-specific data can be generated by any of a number of application programs;

converting the application-specific data into an application-independent format to create an application-independent data bundle;

sending the application-independent data bundle over the communications network to the server;

expanding the neutrally-formatted data bundle into structured data at the server, the structured data at the server specifying representations corresponding to the application-specific structure data that originated on the client; and

generating events to modify the web site in communication with the server to create the specified representations.

64. The method of claim 63 wherein the application-independent format utilizing an extensible markup language.

65. The method of claim 64 wherein the application-specific data includes structured data.

66. The method of claim 1 comprising the further step of maintaining back references from the structured data at the server to the corresponding application-specific data at the client.

67. A method for replicating, on a website in communication with a server, representations of application-specific structured data originating on a client, the method comprising the steps of:

activating an Internet connection between the client and the server;

activating a browser program at the client, the browser program including a GUI;

sending to the client via the Internet a hypertext document containing information representative of a drop zone displayable on the GUI, for receiving, via a GUI drag-and-drop operation, a file;

receiving, via the drop zone, an application-specific file containing a data object characteristics of an application program used to create the file;  
converting the application-specific file into application-independent data;  
sending the application-independent data over a connection to the server;  
expanding the application-independent data into structured data at the server, the structured data at the server specifying an information structure corresponding to information structures specified by the application-specific file; and  
generating an event, based on the structured data at the server, to modify the web site in communication with the server to create the specified information structure.

68. The method of claim 67 wherein a plurality of files are received via the drop zone.

69. The method of claim 67 wherein the application-specific files at the client can be generated by any of a plurality application programs.

70. The method of claim 69 further including the step of maintaining back references from the structured data at the server to the corresponding application-specific file at the client.

71. The method of claim 67 wherein the website has a set of associated users, who are notified when the website is modified.

72. The method of claim 71 wherein the users are notified by e-mail.

73. A method for replicating, on a website in communication with a server, representations of application-specific structured data originating on a client, the method comprising the steps of:

obtaining application-specific data at the client;  
passing the application-specific data to two or more bidding data processor elements;  
causing each of the bidding processors to interpret the application-specific data;

causing each of the bidding processors to generate a bid representative of the ability of the submitting processor to convert the application-specific data into application-independent data;

selecting, on the basis of the generated bids, one of the bidding processors to convert the application-specific data;

causing the selected bidding component to convert the application-specific data into application-independent data;

sending the application-independent data over a connection to the server;

expanding the application-independent data into structured data at the server, the structured data specifying representations corresponding to the application-specific data originating on the client; and

generating an event, based on the structured data at the server, to modify the web site in communication with the server to create the specified information structure.

74. The method of claim 73 including the further step of determining which bidding processors are registered on the client.

75. The method of claim 74 including the further steps of  
determining whether restrictions are associated with the application-specific data;  
and  
passing, to the bidding components, the restrictions with the application-specific data.

76. The method of claim 75, wherein the bid value is based on any of file extension, file content, or restrictions.

77. The method of claim 76, wherein the application-independent data is structured in accordance with an extensible markup language.

78. The method of claim 77 wherein the markup language comprises XML.

79. The method of claim 78 including the further steps of:  
extracting structure information from the application-specific data; and  
sending to the server the structure information along with the application-independent data.
80. The method of claim 78 further including:  
obtaining structure information from a human user at the client;  
sending to the server the structure information along with the application-independent data.
81. The method of claim 80 comprising the further steps of:  
requesting, by generating a request signal at the client, a new data processor element if none of the existing data processor elements is capable of converting the application-specific data into application-independent data; and  
registering the new processor element at the client.
82. The method of claim 81 comprising the further step of utilizing the new processor element to convert the application-specific data into application-independent data.
83. A method of transferring application-generated structured data from a client to a server, such that information structures specified by the data can be automatically expanded into corresponding information structures displayable by a website in communication with the server, the client having a set of Data Clients, the method including the steps of:  
providing an interface at the client for accepting application-generated structured data,  
selecting a Data Client to process the application-generated structured data;  
converting the accepted application-generated structured data, by the selected Data Client, into a source-independent format, thereby to generate a source-independent data package,  
transmitting to the server the source-independent data package,

receiving at the server the source-independent data package, and  
creating or modifying on the server, in accordance with the received source-independent data package, information structures corresponding to the  
information structures specified by the application-generated structured data.

84. The method of claim 83 wherein new Data Clients can be added and registered with the system, whereby the set of Data Clients is extended.

85. The method of claim 84 wherein the client interface accepts a plurality of application-generated structured data sets.

86. The method of claim 85 wherein the plurality of application-generated structured data sets may be generated by a plurality of applications.

87. The method of claim 85 wherein a plurality of Data Clients are selected to convert application-specific structured data, each Data Client being assigned one or more application-specific structured data sets.

88. The method of claim 83 comprising the further step of generating a thumbnail image representative of at least a portion of the information structure.

89. The method of claim 88 comprising the further step of generating a thumbnail image representative of a data object.

90. A system for transferring application-generated structured data from a client to a server, such that information structures specified by the data can be automatically expanded into corresponding information structures displayable by a website in communication with the server, the system including:

an interface at the client for accepting application-generated structured data,

a converter element that converts into a source-independent format the accepted application-generated structured data, thereby to generate a source-independent data package,

a communications channel that transmits to the server the source-independent data package,

elements at the server that receive the source-independent data package, and

elements that create or modify on the server, in accordance with the received source-independent data package, information structures corresponding to the information structures specified by the application-generated structured data.